UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/789,140	02/27/2004	Daryl B. Olander	ORACL-01403US0	9268

23910 7590 10/17/2008

FLIESLER MEYER LLP
650 CALIFORNIA STREET
14TH FLOOR
SAN FRANCISCO, CA 94108

EXAMINER	
BELOUSOV, ANDREY	

ART UNIT	PAPER NUMBER
2174	

MAIL DATE	DELIVERY MODE
10/17/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)
	10/789,140	OLANDER ET AL.
	Examiner	**Art Unit**
	ANDREY BELOUSOV	2174

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 September 2008*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-70* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-70* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date *9/29/2008*.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

This action is in responsive to the filing of 9/29/2008. Claims 1-70 are pending and have been considered below.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-70 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hunter (Java™ Servlet Programming by Jason Hunter, Copyright (c) 2001, 1998 O'Reilly & Associates, Inc.) in view of Griffin (2003/0126558.)

Claim 1: Hunter discloses a method for building a representation of a graphical user interface (GUI), comprising:

a. generating a class (e.g. HolisticCounter), the class being an object oriented programming language class (pg. 40-41);
b. generating a first representation of the GUI (class instance, pg. 41), wherein the class can produce a second representation GUI (e.g. another class instance, pg. 41) based on the first representation (counter value is modified by number of instances, pg. 41);

c. generating a second representation of the GUI (second instance of HolisticCounter class) from the class, wherein the second representation includes at least one control (e.g. count, classCount, instrances, etc., pg. 41); and

d. wherein the first representation includes hierarchical relationships (public / private scope, visibility, local / nonlocal, pg. 38) among controls, control properties (e.g. static, int, public, class, etc., pg. 41), and control event information (exceptions thrown, responses, pg. 41.)

However, Hunter does not explicitly disclose wherein the hierarchical relationships between controls are defined by a control tree that includes a portal control, a desktop control, a page control and a portlet control.

Griffin discloses a similar JSP method for building a representation of a GUI, including wherein the hierarchical relationships between controls are defined by a control tree (Fig. 4) that includes a portal control (Fig. 402), a desktop control (Fig. 4: Skins, templates, etc.), a page control (Fig. 4: 408) and a portlet control (Fig. 410.) Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the teachings of Hunter for the underlying JSP technology for implementation of a JSP backed GUI, with the teaching of Griffin for a multi-tiered implementation including portal, desktop, page and portlet level controls as defined by a control tree. One would have been motivated to combine the multi-tiered implementation as taught by Griffin, with the underlying JSP technology teaching of Hunter as it would have been a natural design choice for a GUI hierarchy within an object-oriented design (OOD) methodology (on which JSP is based) for an enterprise level portal webpage.

Claim 2: Hunter and Griffin disclose the method of claim 1, further comprising: creating the first representation by parsing a file (compiling, pg. 7).

Claim 4: Hunter and Griffin disclose the method of claim 1 wherein: the second representation is a tree (pg.494.)

Claim 5: Hunter and Griffin disclose the method of claim 1 wherein: the step of generating the class occurs as a result of receiving a request (pg. 15.)

Claim 8: Hunter and Griffin disclose the method of claim 1 wherein: the second representation can be driven through at least one lifecycle stage by an interchangeable lifecycle component (pg. 35, 36.)

Claim 17, 58: Hunter discloses a method and machine readable medium having instructions thereon for building a representation of a graphical user interface (GUI), comprising:

a. generating a representation of the GUI (Fig. 3-2, pg. 42) from metadata (instance of a class, pg. 41), wherein the representation includes at least one control (e.g. count, classCount, instrances, etc., pg. 41);

b. driving the representation through at least one lifecycle stage by an interchangeable lifecycle component (pg. 35, 36);

c. wherein the metadata can include at least one of: hierarchical relationships among controls, control properties, and control event information (e.g. static, int, etc., pg. 41); and

d. wherein the representation can be driven through the at least one lifecycle stage (e.g. init, destroy, etc.) by the interchangeable lifecycle component (servlet thread, pg. 35-36);

e. wherein an interface (pg. 266, JDBC SQL-level API, a set of interfaces and classes designed to perform actions against any database) is provided to isolate details of the interchangeable lifecycle component (servlet thread performing any one of: database connection, management, calls, etc., pg. 261-267) from a control container (servlet container, pg. 35) to allow different lifecycle implementations (pg. 266, Fig. 9-2: Oracle, ODBC, Sybase) to be interchangeable (pg. 261, “JDBC is database independent”);

f. wherein the interchangeable lifecycle component runs on at least one processor (pg. 35, 36, it is inherent that ‘execution on a server’ is performed on processor);

However, Hunter does not explicitly disclose wherein the interchangeable lifecycle component further drives a control tree through different lifecycle stages, the control tree includes portal controls, desktop controls, page controls and portlet controls.

Griffin discloses a similar JSP method for building a representation of a GUI, including wherein the interchangeable lifecycle component further drives a control tree

(Fig. 4) through different lifecycle stages, the control tree includes portal controls (Fig. 402, 406, 412), desktop controls (Fig. 4: Skins, templates), page controls (Fig. 4: 408, par 14) and portlet controls (Fig. 410, par 14.) Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the teachings of Hunter for the underlying JSP technology for implementation of a JSP backed GUI, with the teaching of Griffin for a multi-tiered implementation including portal, desktop, page and portlet level controls as defined by a control tree. One would have been motivated to combine the multi-tiered implementation as taught by Griffin, with the underlying JSP technology teaching of Hunter as it would have been a natural design choice for a GUI hierarchy within an object-oriented design (OOD) methodology (on which JSP is based) for an enterprise level portal webpage.

Claim 30: Hunter discloses a system for building a representation of a graphical user interface (GUI), comprising:

a. a first component operable (JVM, pg. 35-36) to produce a second component (servlet object instance, pg. 35-36) and a metadata representation of the GUI (instance of a class, pg. 41);

b. the second component operable to produce a hierarchical representation of the GUI based on the metadata (pg. 42), wherein the representation includes at least one control (e.g. count, classCount, instrances, etc., pg. 41);

c. wherein the metadata can include at least one of: hierarchical relationships among controls, control properties, and control event information (e.g. static, int, etc., pg. 41); and

d. wherein the representation can be driven through at least one lifecycle stage by an interchangeable lifecycle component (servlet thread, pg. 35-36);

e. wherein an interface (pg. 266, JDBC SQL-level API, a set of interfaces and classes designed to perform actions against any database) is provided to isolate details of the interchangeable lifecycle component (servlet thread performing any one of: database connection, management, calls, etc., pg. 261-267) from a control container (servlet container, pg. 35) to allow different lifecycle implementations (pg. 266, Fig. 9-2: Oracle, ODBC, Sybase) to be interchangeable (pg. 261, “JDBC is database independent”); and

f. wherein the interchangeable lifecycle component runs on at least one processor (pg. 35, 36, it is inherent that ‘execution on a server’ is performed on processor.)

However, Hunter does not explicitly disclose wherein the interchangeable lifecycle component further drives a control tree through different lifecycle stages, the control tree includes portal controls, desktop controls, page controls and portlet controls.

Griffin discloses a similar JSP method for building a representation of a GUI, including wherein the interchangeable lifecycle component further drives a control tree (Fig. 4) through different lifecycle stages, the control tree includes portal controls (Fig. 402, 406, 412), desktop controls (Fig. 4: Skins, templates), page controls (Fig. 4: 408, par 14) and portlet controls (Fig. 410, par 14.) Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the teachings of Hunter for the underlying JSP technology for implementation of a JSP backed GUI, with the teaching of Griffin for a multi-tiered implementation including portal, desktop, page and portlet level controls as defined by a control tree. One would have been motivated to combine the multi-tiered implementation as taught by Griffin, with the underlying JSP technology teaching of Hunter as it would have been a natural design choice for a GUI hierarchy within an object-oriented design (OOD) methodology (on which JSP is based) for an enterprise level portal webpage.

Claim 44: Hunter discloses a system comprising:

a. a means (pg. 35-36 JVM) for generating a first representation (e.g. servlet object instance, pg. 35-36) of a graphical user interface (GUI);

b. a means for generating a second representation of the GUI from the first representation (class instance, pg. 41), wherein the second representation includes at least one control (e.g. count, classCount, instrances, etc., pg. 41);

c. wherein metadata (pg. 40, HolisticCounter class) can include at least one of: hierarchical relationships among controls, control properties, and control event information (e.g. static, int, etc., pg. 41); and

d. wherein the second representation can be driven through at least one lifecycle stage by an interchangeable lifecycle component (servlet thread, pg. 35-36);

e. wherein an interface (pg. 266, JDBC SQL-level API, a set of interfaces and classes designed to perform actions against any database) is provided to isolate details of the interchangeable lifecycle component (servlet thread performing any one of: database connection, management, calls, etc., pg. 261-267) from a control container (servlet container, pg. 35) to allow different lifecycle implementations (pg. 266, Fig. 9-2: Oracle, ODBC, Sybase) to be interchangeable (pg. 261, “JDBC is database independent”); and

f. wherein the interchangeable lifecycle component runs on at least one processor (pg. 35, 36, it is inherent that ‘execution on a server’ is performed on processor.)

However, Hunter does not explicitly disclose wherein the interchangeable lifecycle component further drives a control tree through different lifecycle stages, the control tree includes portal controls, desktop controls, page controls and portlet controls.

Griffin discloses a similar JSP method for building a representation of a GUI, including wherein the interchangeable lifecycle component further drives a control tree (Fig. 4) through different lifecycle stages, the control tree includes portal controls (Fig. 402, 406, 412), desktop controls (Fig. 4: Skins, templates), page controls (Fig. 4: 408, par 14) and portlet controls (Fig. 410, par 14.) Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the teachings of Hunter for the underlying JSP technology for implementation of a JSP backed GUI, with the teaching of Griffin for a multi-tiered implementation including portal, desktop, page and portlet level controls as defined by a control tree. One would have been motivated to combine the multi-tiered implementation as taught by Griffin, with the underlying JSP technology teaching of Hunter as it would have been a natural design choice for a GUI hierarchy within an object-oriented design (OOD) methodology (on which JSP is based) for an enterprise level portal webpage.

Claim 18, 31, 45, 59: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 17, 30, 44, and 58, respectively, further comprising: creating the metadata by parsing a file (compiling, pg. 7.)

Claim 19, 33, 60: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 17, 30, and 58, respectively, wherein: the step of generating the metadata representation occurs as a result of receiving a request (pg. 15.)

Claim 9, 22, 36, 50, 63: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 1, 17, 30, 44, and 58, respectively, wherein: the at least one control has an interchangeable persistence mechanism (pg. 37, 216, 384, 582.)

Claim 10, 23, 37, 51, 64: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 1, 17, 30, 44, and 58, respectively, wherein: the at least one control can render itself according to a theme (Tea Templates, pg. 433.)

Claim 11, 24, 38, 52, 65: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 1, 17, 30, 44, and 58, respectively, wherein: one of the at least one controls can interact with another of the at least one controls (pg. 35.)

Claim 12, 25, 39, 53, 66: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 1, 17, 30, 44, and 58, respectively, wherein: one of the at least one controls can advance through the at least one lifecycle stage in parallel with another of the at least one controls (pg. 35.)

Claim 13, 26, 40, 54, 67: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 8, 17, 30, 44, and 58, respectively, wherein:

a. the at least one lifecycle stage is one of: init, load state, create child controls, load, raise events, pre-render, render, save state, unload and dispose (pg. 35, 43); and
b. wherein the lifecycle stage is part of a dynamically configurable lifecycle (pg. 35.)

Claim 15, 28, 42, 56, 69: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 1, 17, 30, 44, and 58, respectively, wherein: the at least one control can raise events and respond to events (pg. 580-583.)

Claim 16, 29, 43, 57, 70: Hunter and Griffin disclose the method, system, and a machine readable medium having instructions thereon, of claim 1, 17, 30, 44, and 58, respectively, wherein: the at least one control can be one of: Book, Page, Window, Menu, Layout, Portlet, Theme, Placeholder, Shell, LookAndFeel, Desktop, Body, Footer, Header, Head, Titlebar, ToggleButton, Treeview, TreeViewWithRadioButtons (pg. 347-353.)

Claim 47: Hunter and Griffin disclose the system of claim 44, further comprising: the means for accepting a request (pg. 15.)

Claim 3, 32, 46: Hunter and Griffin disclose a method, and a system of claims 2, 31 and 45, respectively, wherein the file is a JavaServer Pages (JSP) file (514-515.)

Claim 6, 20, 34, 48, 61: Hunter and Griffin disclose a method, system and a machine readable medium having instructions thereon, of claims 5, 19, 33, 47 and 60, respectively, wherein the request is a hypertext transfer protocol request (HTTP); and the request originates from a web browser (pg. 15.)

Claim 7, 21, 35, 49, 62: Hunter and Griffin disclose a method, system and a machine readable medium having instructions thereon, of claims 1, 17, 30, 44 and 58, respectively, further comprising: providing a response to a web browser (pg. 15, 42.)

Claim 14, 27, 41, 55, 68: Hunter and Griffin disclose a method, system and a machine readable medium having instructions thereon, of claims 7, 21, 35, 49 and 62, respectively, wherein the response is a hypertext transfer protocol (HTTP) response (pg. 15, 42.)

Response to Arguments

3. Applicant's arguments with respect to claim1, 17, 30, 44, and 58 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Andrew Belousov whose telephone number is (571) 270-1695. The examiner can normally be reached on Mon-Fri (alternate Fri off) EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner’s supervisor, Stephen Hong can be reached on (571) 272-4124. The fax phone number for the organization where this application or proceeding is assigned is 571-273-3800.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

AB

/Steven P Sax/
Primary Examiner, Art Unit 2174

October 10, 2008